LIST OF CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

Claims 1-4 (cancelled)

5.   (Currently amended) A method for scheduling events in a computer processing system, comprising:

identifying queues, each of the queues associated with a corresponding priority, each of the queues including events;

defining a data structure with a root level having a node group, the node group having k number of nodes, each of the k number of nodes sharing a pointer, each of the k number of nodes stored contiguously in memory, wherein the k number is equal to a number of multiple queues;

associating the queues with respective nodes of the data structure;

assigning a value representing the corresponding priority to the respective nodes;

determining a priority between the respective nodes based on respective values representing the corresponding priority to the respective nodes; ~~and~~

selecting one of the events corresponding to a node having a highest priority for transmission to a processing resource<ins>; and</ins>

<ins>processing the selected one of the events at the processing resource prior to remaining events</ins>.

6.   (previously presented) The method of claim 5, further comprising:

rescheduling the node having the highest priority after selection.

7.   (previously presented) The method of claim 6, wherein the method operation of
rescheduling the node having the highest priority after selection includes,

determining if the node having the highest priority will be empty after selection.

8.   (previously presented) The method of claim 7, further comprising:

if the node having the highest priority will be empty after selection, then the method
includes,

removing the value representing the corresponding priority from the node having
the highest priority.

9.   (previously presented) The method of claim 7, further comprising:

if the node having the highest priority will not be empty after selection, then the method
includes,

retaining the value representing the corresponding priority from the node having the
highest priority, thereby enabling rescheduling of the node having the highest priority after
selection.

10.   (previously presented) The method of Claim 5, further comprising:

resolving conflicts between respective nodes assigned a same value by rotating a pointer among the respective nodes assigned the same value.

Claims 11-21 (cancelled)

22.   (currently amended) A computer readable medium having program instructions for scheduling events in a computer processing system, comprising:

program instructions for identifying queues, each of the queues associated with a corresponding priority, each of the queues including events;

program instructions for defining a data structure with a root level having a node group, the node group having k number of nodes, each of the k number of nodes sharing a pointer, each of the k number of nodes stored contiguously in memory, wherein the k number is equal to a number of multiple queues;

program instructions for associating the queues with respective nodes of the data structure;

program instructions for assigning a value representing the corresponding priority to the respective nodes;

program instructions for determining a priority between the respective nodes based on respective values representing the corresponding priority to the respective nodes; and

program instructions for selecting one of the events corresponding to a node having a highest priority for transmission to a processing resource; and

program instructions for processing the selected one of the events at the processing resource prior to remaining events.

23.   (previously presented) The computer readable medium of claim 22, further comprising:

rescheduling the node having the highest priority after selection.

24.   (previously presented) The computer readable medium of claim 23, wherein the program instructions for rescheduling the node having the highest priority after selection includes,

determining if the node having the highest priority will be empty after selection.

25.   (previously presented) The computer readable medium of claim 24, further comprising:

if the node having the highest priority will be empty after selection, then the computer readable medium includes,

program instructions for removing the value representing the corresponding priority from the node having the highest priority.

26.   (previously presented) The computer readable medium of claim 24, further comprising:

if the node having the highest priority will not be empty after selection, then the computer readable medium includes,

program instructions for retaining the value representing the corresponding priority from the node having the highest priority, thereby enabling rescheduling of the node having the highest priority after selection.

27.   (original) The computer readable medium of claim 22, further comprising:

program instructions for resolving conflicts between respective nodes assigned a same value by rotating an additional pointer among the respective nodes assigned the same value.